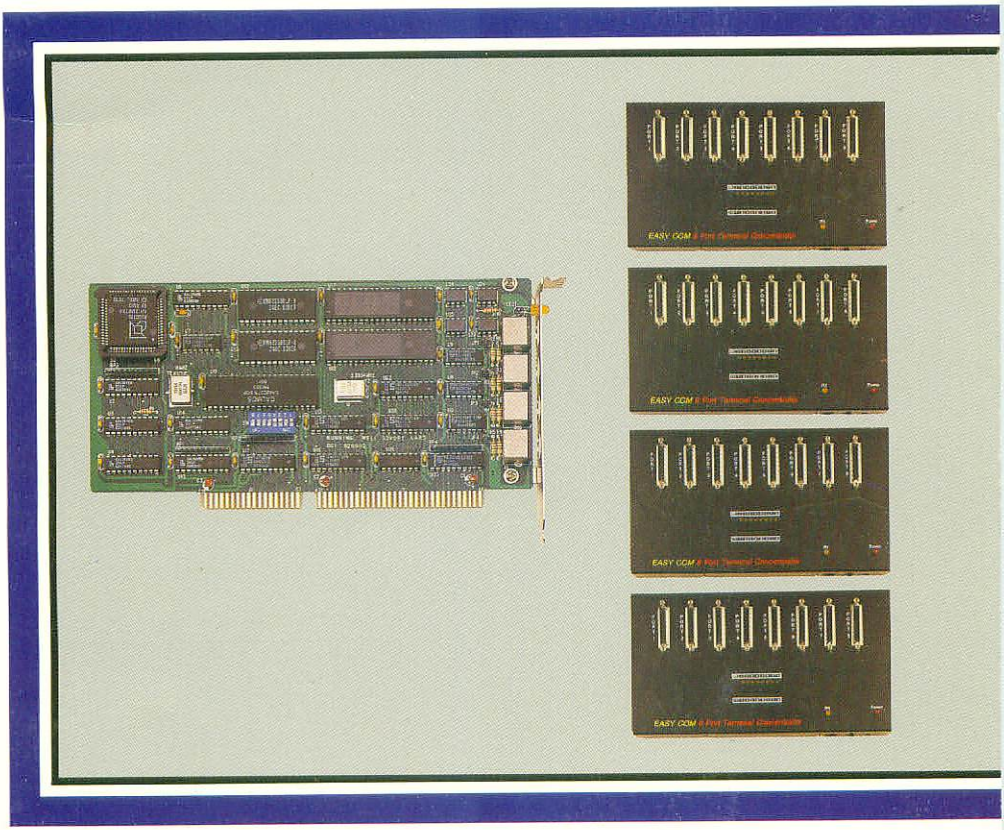
EZ COM USER MANUAL
8 * 4 PORT INTELLIGENT SERIAL ADAPTER & CONCENTRATOR



For XENIX/UNIX USERS

In this manual, page 17 and page 19 have some mistakes. Please revise them before you install the driver to XENIX/UNIX operating system. Sorry, that let you inconvenient.

Page	Mistake	Revised
17	#doscp a:/ezcom.z /ezcom.tar.z	#doscp a:/ezcom.z /ezcom.tar.Z
17	#uncompress /ezcom.tar.z	#uncompress /ezcom.tar.Z
19	#doscp a:/ezcom.z /ezcom.tar.z	#doscp a:/ezcom.z /ezcom.tar.Z
19	#uncompress /ezcom.tar.z	#uncompress /ezcom.tar.Z

EZ COM USER MANUAL

8*4 PORTS INTELLIGENT SERIAL ADAPTER and CONCENTRATOR

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION 1

CHAPTER 2 UNPACKING INFORMATION 4

CHAPTER 3 SYSTEM REQUIREMENTS 5

CHAPTER 4 HARDWARE INSTALLATION 6

CHAPTER 5 SWITCH SETTINGS 7

5.1 INTRODUCTION 7
5.2 SELECTION OF SWITCH SETTINGS 7

APPENDIX A CABLING INFORMATION 10

A.1 EZ COM WITH TERMINAL CONCENTRATOR 10
A.2 8-PIN MINI-DIN CONNECTOR 10
A.3 DB25 CONNECTORS - RS232 TYPE 11
A.4 CABLE CONNECTIONS - RS232 TYPE 12
A.5 RJ-11 CONNECTOR 12
A.6 RJ-45 CONNECTOR 13

APPENDIX B NULL MODEM AND MODEM CABLE 14

B.1 NULL MODEM CONNECTIONS: RS232 14
B.2 MODEM CONNECTIONS 15

APPENDIX C SCO XENIX CONFIGURATION 16

C.1 HARDWARE CONFIGURATION 16
C.2 INSTALLATION FOR SCO XENIX 16

APPENDIX D SCO UNIX SYSTEM V, AT&T UNIX SYSTEM V, INTERACTIVE UNIX SYSTEM V 18

D.1 HARDWARE CONFIGURATION 18
D.2 SOFTWARE INSTALL PROCEDURE 18

APPENDIX E MS/DOS CONFIGURATION 21

E.1 DEVICE DRIVER INSTALLATION 21
E.2 DEVICE NAME 23
E.3 COMMUNICATION PARAMETERS SET UP 27

APPENDIX F PROGRAMMING TECHNIQUES UNDER MS/DOS 32

F.1 HOW TO USE DEVICE DRIVER 32

F.2 EZ COM LIBRARY 37
F.3 EXAMPLES 45
F.4 I/O CONTROL VIA SOFTWARE INTERRUPT 46

APPENDIX G PC COM VIRTUAL TERMINAL UTILITIES 54

G.1 VALID CONTROL SEQUENCE OF VT 54
G.2 EXTENDED GRAPHICS CAPABILITY 55
G.3 HOW TO SETUP PC COM DRIVER TO RUN VT 56
G.4 HOW TO EXECUTE VT.EXE 59
G.4.1 DOWNLOAD DATA FROM HOST 60
G.4.2 UPLOAD DATA TO HOST 61
G.4.3 SETUP 61
G.5 EXAMPLES 62

APPENDIX H TRANSPARENT PRINTING UNDER UNIX 64

H.1 DEVICE NAME 64
H.2 TRANSPARENT PRINTING CONTROL 64
H.3 HOW TO USE TRANSPARENT PRINTER 65

APPENDIX I TECHNICAL REFERENCE INFORMATION 67

APPENDIX J IMPORTANT TIPS 68

APPENDIX K WARRANTY INFORMATION 69

K.1 COPYRIGHT 69
K.2 WARRANTY INFORMATION 69

CHAPTER 1 INTRODUCTION

The EZ COM intelligent serial package is the Corollary 8*4 adapter and 8/tc concentrator compatible product, which contains one intelligent serial adapter and one terminal concentrator. Each terminal concentrator provides eight asynchronous serial communication ports (RS232), which transmits communication signals by using the cable with an 8-pin mini-DIN connector to intelligent serial adapter. The hardware configuration of intelligent serial adapter and 8 port terminal concentrators are shown in the belows. User may use this configuration to link the computer and serial peripheral devices such as serial printers, plotters, etc.

Each EZ COM intelligent serial adapter can be used to connect up to four terminal concentrators, total 32 RS232 ports are provided. Up to four EZ COM intelligent serial adapters can be inserted into one computer system, so that it supports 128 ports in one computer system.

The advantages of using EZ COM intelligent serial adapter and 8 port terminal concentrators are that when you construct the communication link from PC to peripheral devices, you only need use one inexpensive cable line to replace the eight RS232 cables, it extends communication distance and reduces cabling cost. Each intelligent serial adapter contains four high speed communication links, so that it can connect to four terminal concentrators, which only needs to use four inexpensive cables rather than 32 RS232 cables.

The EZ COM intelligent adapter is particular suited to facilitate the connection of terminals (VDUs) in multi- user operating systems. Each package is supplied complete with one terminal concentrator containing eight female DB25 connectors.

If extra concentrators are needed, please order the concentrator item.

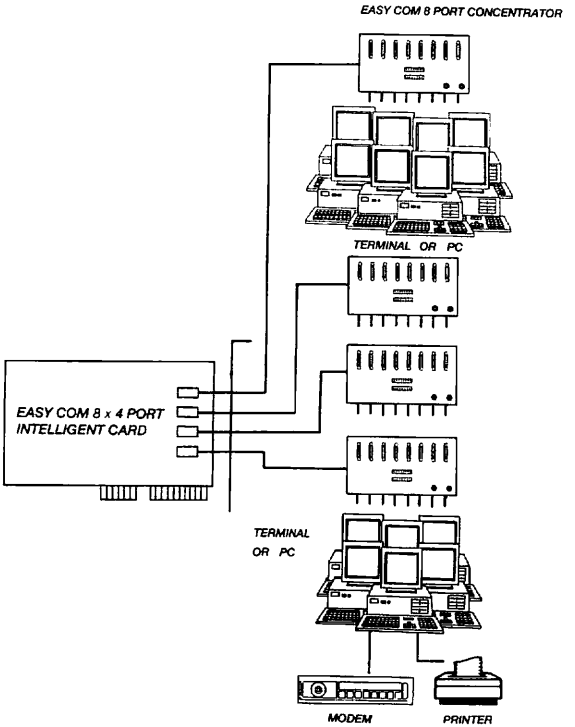
The EZ COM intelligent serial board may be installed in any IBM PC/AT, PC/386, PC/486 or hardware compatible system. To accommodate a variety of operating systems, the distribution diskettes contains many device drivers for the operating system you need.

The architecture of EZ COM intelligent package provides the distribution multiprocessor I/O system, both intelligent serial adapter and terminal concentrator use HIGH intelligent CPU main processor which handles the send and receive functions between the intelligent serial adapter and terminal concentrators, it saves the workload of the host computer.

The features of the EZ COM intelligent package are:

- CPU is provided on both intelligent serial adapter and terminal concentrator.
- Each terminal concentrator contains eight RS232 ports for asynchronous communications.
- Up to four terminal concentrators are connected to one EZ COM serial adapter, total 32 ports are provided.
- Up to four EZ COM serial adapters can be inserted into one computer system, total 128 ports are provided.
- Suitable for XENIX/UNIX, MS/DOS.
- IBM PC/AT, PC/386, PC/486 hardware compatible.
- Only one inexpensive cable line with 8- pin mini-DIN connector to replace eight RS232 asynchronous communication cables.
- Maximum transmission distance over 1000 ft (300m).
- Baud rate selectable for each individual port, and the maximum is 38400.
- Software handshaking (XON/XOFF) and hardware handshaking (RTS/CTS) selectable.

- 9V power adapter for terminal concentrator.
- Corollary 8*4 adapter and 8/tc concentrator compatible.
- Provides transparent printing on terminal.



CHAPTER 2 UNPACKING INFORMATION

Check that your EZ COM intelligent package includes the following items:

- EZ COM intelligent adapter.
- EZ COM 8 port terminal concentrator.
- 9V power adapter.
- EZ COM cable line with 8-pin mini-DIN connector.
- User manual.
- EZ COM software and device drivers for operating systems.
- Warranty form.

CHAPTER 3 SYSTEM REQUIREMENTS

Before installing your EZ COM intelligent serial adapter and 8 port terminal concentrators, make sure that:

- The host computer is an IBM PC/AT, PC/386, PC/486 or compatible.
- The one switch bank and four jumpers of EZ COM adapter and one switch bank of terminal concentrator are correctly configured to coincide with the operating system you are using.
- If the distance between the adapter and terminal concentrator is more than 3m , please use power adapter.
- For short distance, the power of each adapter only provides two terminal concentrators.

CHAPTER 4 HARDWARE INSTALLATION

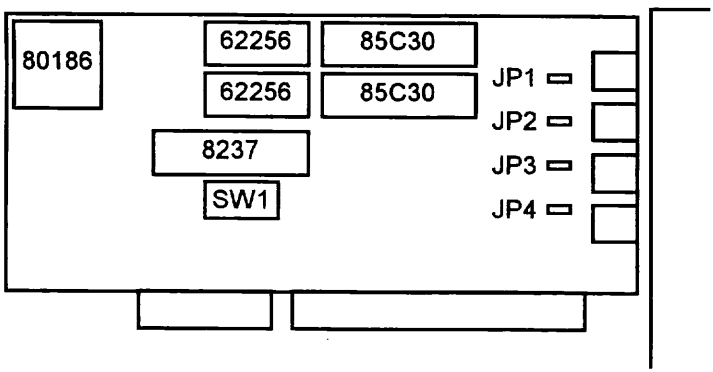
Your EZ COM intelligent serial adapter is designed to be inserted in any available slot in your PC/AT, PC/386, PC/486 or compatible. In order to gain access to the expansion slots, follow the steps listed below:

1. **Turn off all power to your computer and all peripheral devices before installing your EZ COM intelligent serial adapter.**
2. **Remove the cover of the computer.**
3. **Insert the preconfigured EZ COM intelligent serial adapter into any available slot. Make sure the adapter is firmly seated in the chosen slot.**
4. **Replace the cover of the computer.**
5. **Route the cable line with 8-pin mini-DIN into both terminal concentrator and intelligent serial adapter.**
6. **Connect peripheral devices connector to terminal concentrator.**
7. **Plug the power cable into wall outlet if needed.**

CHAPTER 5 SWITCH SETTINGS

5.1 Introduction

The one switch block and four jumpers on the EZ COM adapter and one switch on the terminal concentrator must be configured correctly in accordance with the operating system you are using.



Switch on adapter

Determines the address of the BIOS and I/O buffer.

Jumpers on adapter

Determines whether the power of terminal concentrator is provided by adapter.

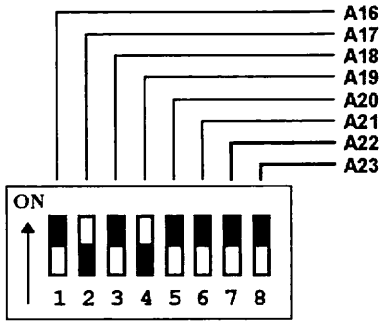
Switch on terminal concentrator

Determines the power source.

If more than one adapter are installed, please set different segment address, and do not conflict with the other adapters such as network card, VGA card, ... etc.

5.2 Selection of Switch Settings

1. Switch on adapter



The switch is used to determine segment address of on board BIOS, I/O buffer, ... etc. when the corresponding switch is set to off, it means 1, otherwise set to on, it means 0. SW-8 represents address line A23, SW-7 represents address line A22, ... etc. You may set on and off configuration to the segment address you require. The figure shown in the above is 0A000:0000.

Each adapter occupies 64K bytes to map to the address of host, **PLEASE CHECK YOUR PC SYSTEM CONFIGURATION TO AVOID CONFLICT WITH THE OTHER ADAPTER.**

2. JP1, JP2, JP3, and JP4

The JP1, JP2, JP3, and JP4 are used to determine whether the power is supported from adapter to terminal concentrator. When short the jumper, the power is supported from adapter, otherwise when open the jumper, the power of terminal concentrator is supported from extra power adapter. JP1 is used to determine the power of the first terminal concentrator, and JP2 is used to determine the power of the second terminal concentrator, ... etc.

3. Switch on terminal concentrator

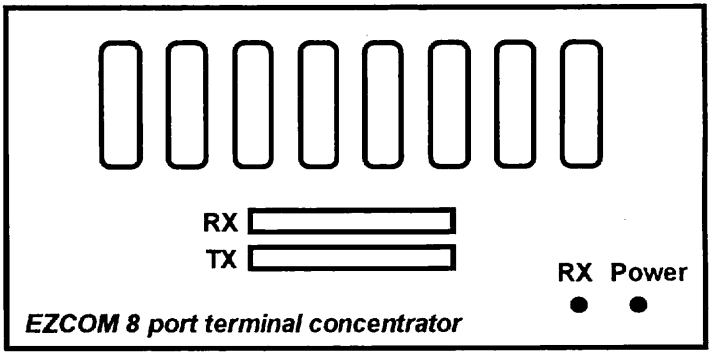
The switch on the terminal concentrator is used to determine whether the power is supported from intelligent adapter. When set to on, the power source comes from power adapter, otherwise when set to off, the power source comes from intelligent adapter.

Please note that, if you select power source from intelligent adapter, each adapter only support two terminal concentrators. If the distance between intelligent and terminal concentrator more than 3m , you need use power adapter to avoid lost of data.

APPENDIX A CABLING INFORMATION

A.1 EZ COM with Terminal Concentrator

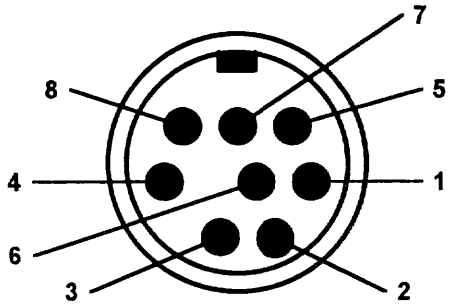
EZ COM with the terminal concentrator provides an RS232C interface with DB25 female connectors for each port.



A.2 8-pin mini-DIN Connector

An 8-pin mini-DIN connector with cable is used to connect host and terminal concentrator. The signals of the connector is shown in the follows.

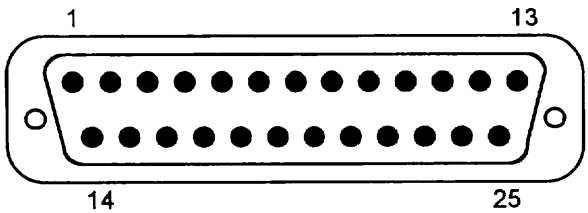
Signal no.	Signal name
1	data+
2	data-
3	+5V
4	GND
5	GND
6	+5V
7	+5V
8	GND



A.3 DB25 Connectors - RS232 Type

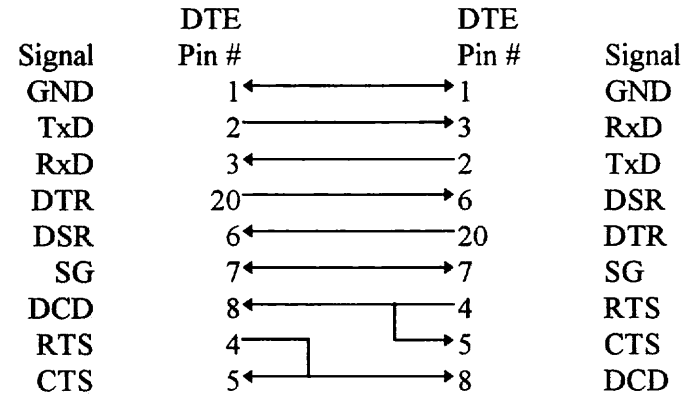
The signal assignments for a standard DB25 connector are shown below:

DB25 Pin #	RS-232C Signal Name	Signal Name	Signal Direction
1	Chassis Ground (GND)	AA	Common
2	Transmit Data (TxD)	BA	Output
3	Receive Data (RxD)	BB	Input
4	Request to Send(RTS)	CA	Output
5	Clear to Send (CTS)	CB	Input
6	Data Set Ready (DSR)	CC	Input
7	Signal Ground (SG)	AB	Common
8	Data Carrier Detect (DCD)	CF	Input
20	Data Terminal Ready (DTR)	CD	Output



A.4 Cable connections - RS232 Type

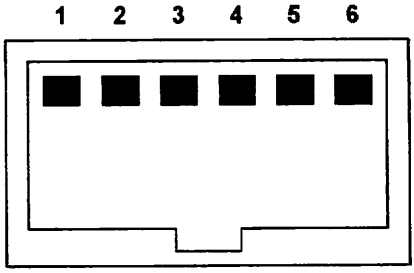
To connect the 8 port from terminal concentrator to other DATA TERMINAL EQUIPMENT (DTE) devices, we recommend using a DTE to DTE connection as shown below:



A.5 RJ-11 Connector

An optional DB25 to RJ-11 module jack is provided for each RS232 port. The signal assignment of RJ-11 are shown in the follows.

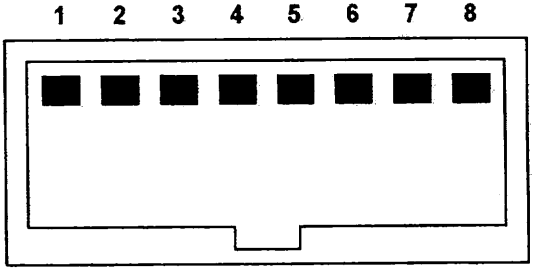
Pin No.	Signal Name
1	Shield Ground
2	Received Data
3	Transmit Data
4	Signal Ground
5	Data Set Ready
6	no connect



A.6 RJ-45 Connector

An optional DB25 to RJ-45 module jack is provided for each RS232 port. The signal assignment of RJ-45 are shown in the follows.

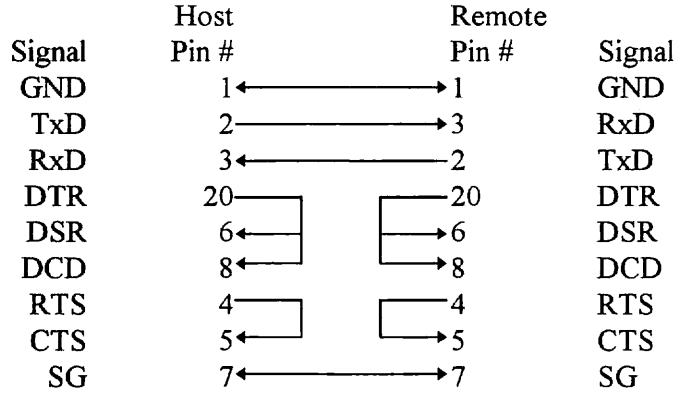
Pin No.	Signal Name
1	Ground
2	Request to Send
3	Data Terminal Ready
4	Transmit Data
5	Received Data
6	Carrier Detect
7	Clear to Send
8	Data Set Ready



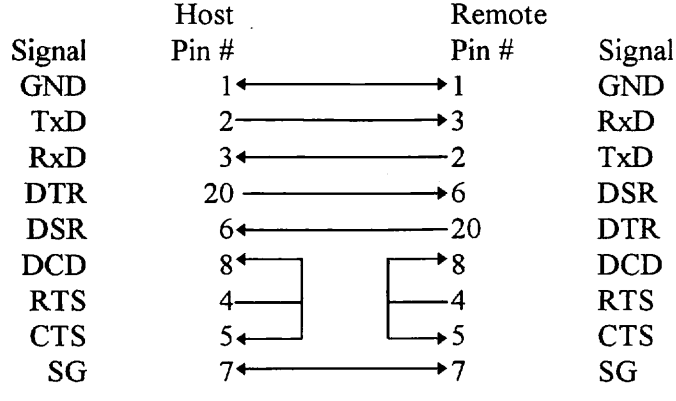
APPENDIX B NULL MODEM AND MODEM CABLE

B.1 Null Modem Connections: RS232

If the software supplier or operating system does not specify a particular cable configuration, we recommend you use the following "null modem" cable when XON/XOFF is utilized.



If hardware handshaking is necessary, use the following cable:



Some serial devices have the buffer control signal on pin 19, in which case pin 6 on the host is connected to pin 19 on the remote device.

B.2 Modem Connections

A straight through cable is required, e.g. pin 2 to pin 2, pin 3 to pin 3, etc.

(DTE) Signal	Host Pin #		Remote Pin #	(DCE) Signal
GND	1	⟷	1	GND
TxD	2	⟶	2	TxD
RxD	3	⟵	3	RxD
RTS	4	⟶	4	RTS
CTS	5	⟵	5	CTS
DSR	6	⟵	6	DSR
SG	7	⟷	7	SG
DCD	8	⟵	8	DCD
DTR	20	⟶	20	DTR

APPENDIX C SCO XENIX CONFIGURATION

C.1 Hardware Configuration

Before insert the EZ COM adapter into computer slot, please determine the on board segment address. You may select range from A0000 to D0000 which below 1M address, also select above 1M address range from C00000 to EFFFFF. Each board occupies 64K bytes memory, and do not forget select different address for more than one board.

The device name is /dev/ttyXYZ, where X is a letter from "a" to "d" corresponds to four adapters, Y is a number from 1 to 4 corresponds to four terminal concentrator of adapter, and Z is a number from 1 to 8 corresponds to eight ports. For example, the /dev/ttya36 is the port 6 of the third terminal concentrator of the first adapter. The MODEM device name is /dev/ttyA36.

During the software installation, you must answer IRQ number; you need not select IRQ number from hardware, however it needs to emerge interrupt from software. Please select the different IRQ number for all boards.

C.2 Installation for SCO XENIX

After the hardware configuration is set, please install the software drivers. The installation procedure is described as follows:

1. **Login as a root user.**
2. **Backup system kernel please type**
   # cp /xenix /xenix.old
   (where # is the system prompt)

3. **Insert distribution diskette labeled SCO XENIX System V into floppy disk drive A:, then copy the files from the distribution diskette to root directory.**
   # cd /
   # doscp a:/ezcom.z /ezcom.tar.z
4. **To install device drivers, please type:**
   # uncompress /ezcom.tar.z
   # tar xvf /ezcom.tar
   # cd /ezcom
   # ./ecmenu
   The setup program will ask you some questions to setup your kernel.
5. **Reboot the system. Now, your new XENIX system which includes device drivers is activated.**
6. **Connect each terminal to DB25 connector.**

NOTE

1. **If the new system fails to reboot, boot the original system using the command:**
   # /xenix.old
2. **To remove driver from xenix, please type the followings and select *uninstall* function.**
   # cd /ezcom
   # ./ecmenu
3. **After installation, the system enable each port automatically, you may use disable command to disable your tty port or use enable to enable your tty port. To enable tty please type**
   # enable /dev/ttya11
   # enable /dev/ttya12
   ...

To disable tty please type
   # disable /dev/ttya11
   # disable /dev/ttya12
   ...

APPENDIX D SCO UNIX SYSTEM V, AT&T UNIX SYSTEM V, INTERACTIVE UNIX SYSTEM V Release 3.2, 4.0 CONFIGURATION

D.1 Hardware Configuration

Before insert the EZ COM adapter into computer slot, please determine the on board segment address. You may select range from A0000 to D0000 which below 1M address, also select above 1M address range from C00000 to EFFFFF. Each board occupies 64K bytes memory, and do not forget select different address for more than one board.

The device name is /dev/ttyXYZ, where X is a letter from "a" to "d" corresponds to four adapters, Y is a number from 1 to 4 corresponds to four terminal concentrator of adapter, and Z is a number from 1 to 8 corresponds to eight ports. For example, the /dev/ttya36 is the port 6 of the third terminal concentrator of the first adapter The MODEM device name is /dev/ttyA36.

During the software installation, you must answer IRQ number; you need not select IRQ number from hardware, however it needs to emerge interrupt from software. Please select the different IRQ number for all boards.

D.2 Software Install Procedure

After the hardware configuration is set, please install the software drivers. The installation procedure is described as follows:

1. **Login as a root user.**
2. **Backup system kernel please type**
   # cp /unix /unix.bck
   (where # is the system prompt)

3. **Insert distribution diskette into floppy disk drive A:, then copy the files from the distribution diskette to root directory.**
   # cd /
   # doscp a:/ezcom.z /ezcom.tar.z
4. **To install device drivers, please type:**
   # uncompress /ezcom.tar.z
   # tar xvf /ezcom.tar
   # cd /ezcom
   # ./ecmenu
   The setup program will ask you some questions to setup your kernel.
5. **Reboot the system. Now, your new UNIX system which includes device drivers is activated.**
6. **Connect each terminal to DB25 connector.**

NOTE

1. **Under the SCO UNIX, if the new system fails to reboot, boot the original system using the command:**
   # /unix.bck
2. **For interactive UNIX, when system is fail for boot, please press return key to halt autoboot, then type**
   # /unix.bck
3. **For AT&T UNIX, if system fail to boot, please modify AUTOBOOT=NO parameter from /etc/default/boot file, then type**
   # /unix.bck
   Please note that, before you install your distribution disk, you need modify AUTOBOOT parameter.
4. **To remove driver from unix, please type the followings and select *uninstall* function.**
   # cd /ezcom
   # ./ecmenu
5. **After installation, the system enable each port automatically. Under the SCO UNIX, you may use disable command to disable your tty port or use enable to enable your tty port. To enable tty please type**

 # enable /dev/ttya11
 # enable /dev/ttya12
...
To disable tty please type
 # disable /dev/ttya11
 # disable /dev/ttya12
...

6. **To disable and enable tty under AT&T UNIX and Interactive UNIX, please use the entty command to enable your tty port, and use the distty command to disable your tty port.**
   # entty ttya11
   # entty ttya12
   ...
   To disable tty please type
   # distty ttya11
   # distty ttya12
   ...

APPENDIX E MS/DOS CONFIGURATION

E.1 Device Driver Installation

The RW.SYS is a device driver for EZ COM intelligent 8 port card. Please add the following line

DEVICE=RW.SYS -v -n32

to your CONFIG.SYS file then power off system and install EZ COM intelligent 8 port card before rebooting system.

Where the option fields are described in the follows.

-v

Verbose the install message, such as IRQ, buffer size, etc.

-q

Shows the install message, only quiet mode.

-a[n]

Where n is hexadecimal for segment address, default is D000. Under the MS/DOS configuration, the segment address must be set below 1M address. If more than one board are installed, repeat this optional switch. (if more than one boards, please set different address).

-n[n]

Where n in the [] is specified number of ports attached to the system. The default value is 8, maximum number is 128.

-b[n]

Where n is the input buffer size, the unit is byte. default value is 512.

-i[n]

Where n is the IRQ number. If more than one boards insert to the system, they use common IRQ. The default value is 11 (decimal).

-c[n]

Determines the device driver name start from which number. The default value is 5. It means the device driver name start from COM5, COM6, COM7, ... etc.

-p[n]

Determines the start port number for BIOS. The default value is 4. NORMALLY, DO NOT USE THIS FIELD, the port number will be defined by device driver automatically, it depends on driver name specified. If you select driver name start from COM5, then the port number is set to 4, if you select driver name start from COM6, then it was set to 5.

-t[n]

Determine time out. The default value is 3 seconds.

The default values of each option field are summary in the follows.

option field	default value
-v	False
-q	False
-a	D000 (hexadecimal)
-n	8
-b	512
-I	11
-c	5
-p	4
-t	3

The default communication parameters for each port are

Baud Rate	9600
Parity	None
Stop Bits	1 Bit
Data Bits	8 Bits

Suppose you have two boards installed in the computer system, and one occupies D000, one occupies A000 segment

address, the number of ports are 64, then in the configuration file you must specify as

DEVICE = RW.SYS -v -aD000 -aA000 -n64

E.2 Device Name

After device driver is installed, the device name is registered by MS/DOS, user may use device name as a file to open, write, read, or use commands to specify I/O redirection easily. For example

type rsetup.c >COM5

Suppose in the RW.SYS setting we set -c[5] and -p[4], then the device name correspond to each port are shown in the follows:

Card Id	Box Id	Port Id	Port No.	Device Name
1	1	1	4	COM5
1	1	2	5	COM6
1	1	3	6	COM7
1	1	4	7	COM8
1	1	5	8	COM9
1	1	6	9	COM10
1	1	7	10	COM11
1	1	8	11	COM12
1	2	1	12	COM13
1	2	2	13	COM14
1	2	3	14	COM15
1	2	4	15	COM16
1	2	5	16	COM17
1	2	6	17	COM18
1	2	7	18	COM19
1	2	8	19	COM20
1	3	1	20	COM21

1	3	2	21	COM22
1	3	3	22	COM23
1	3	4	23	COM24
1	3	5	24	COM25
1	3	6	25	COM26
1	3	7	26	COM27
1	3	8	27	COM28
1	4	1	28	COM29
1	4	2	29	COM30
1	4	3	30	COM31
1	4	4	31	COM32
1	4	5	32	COM33
1	4	6	33	COM34
1	4	7	34	COM35
1	4	8	35	COM36
2	1	1	36	COM37
2	1	2	37	COM38
2	1	3	38	COM39
2	1	4	39	COM40
2	1	5	40	COM41
2	1	6	41	COM42
2	1	7	42	COM43
2	1	8	43	COM44
2	2	1	44	COM45
2	2	2	45	COM46
2	2	3	46	COM47
2	2	4	47	COM48
2	2	5	48	COM49
2	2	6	49	COM50
2	2	7	50	COM51
2	2	8	51	COM52
2	3	1	52	COM53
2	3	2	53	COM54

2	3	3	54	COM55
2	3	4	55	COM56
2	3	5	56	COM57
2	3	6	57	COM58
2	3	7	58	COM59
2	3	8	59	COM60
2	4	1	60	COM61
2	4	2	61	COM62
2	4	3	62	COM63
2	4	4	63	COM64
2	4	5	64	COM65
2	4	6	65	COM66
2	4	7	66	COM67
2	4	8	67	COM68
3	1	1	68	COM69
3	1	2	69	COM70
3	1	3	70	COM71
3	1	4	71	COM72
3	1	5	72	COM73
3	1	6	73	COM74
3	1	7	74	COM75
3	1	8	75	COM76
3	2	1	76	COM77
3	2	2	77	COM78
3	2	3	78	COM79
3	2	4	79	COM80
3	2	5	80	COM81
3	2	6	81	COM82
3	2	7	82	COM83
3	2	8	83	COM84
3	3	1	84	COM85
3	3	2	85	COM86
3	3	3	86	COM87

3	3	4	87	COM88
3	3	5	88	COM89
3	3	6	89	COM90
3	3	7	90	COM91
3	3	8	91	COM92
3	4	1	92	COM93
3	4	2	93	COM94
3	4	3	94	COM95
3	4	4	95	COM96
3	4	5	96	COM97
3	4	6	97	COM98
3	4	7	98	COM99
3	4	8	99	COM100
4	1	1	100	COM101
4	1	2	101	COM102
4	1	3	102	COM103
4	1	4	103	COM104
4	1	5	104	COM105
4	1	6	105	COM106
4	1	7	106	COM107
4	1	8	107	COM108
4	2	1	108	COM109
4	2	2	109	COM110
4	2	3	110	COM111
4	2	4	111	COM112
4	2	5	112	COM113
4	2	6	113	COM114
4	2	7	114	COM115
4	2	8	115	COM116
4	3	1	116	COM117
4	3	2	117	COM118
4	3	3	118	COM119
4	3	4	119	COM120

4	3	5	120	COM121
4	3	6	121	COM122
4	3	7	122	COM123
4	3	8	123	COM124
4	4	1	124	COM125
4	4	2	125	COM126
4	4	3	126	COM127
4	4	4	127	COM128
4	4	5	128	COM129
4	4	6	129	COM130
4	4	7	130	COM131
4	4	8	131	COM132

E.3 Communication Parameters Set Up

Before setup communication parameters, we describe the communication parameters in the following:

baud rate

The transmit speed between two serial ports.

parity

The parity bit is used in error detection. Three different settings are used, which are ODD, EVEN, and NONE parity.

data bits

The number of bits in each transmitted or received serial character. Either 7 or 8 data bits can be specified.

stop bits

The stop bits is used to detect where one character ends and another starts. Either 1 or 2 stop bits can be specified. When stop bits is specified then stop bits are generated or checked in the transmit or receive data respectively.

XON/XOFF

XON/XOFF is software handshaking, which is used to prevent the lost of data while communication between two

serial ports. Suppose we connect two computer named local computer and remote computer, if local computer is too busy to receive data from remote computer, it send XOFF (S/CTRL whose code is 19) to remote computer to notice he is in busy mode. When remote computer receives XOFF, it will stop transmission until the local computer sends XON (Q/CTRL whose code is 17) to continue transmission. The same manner is activated at remote computer.

The syntactic specification of RSETUP is

RSETUP devicename, [baud, parity, data, stop, handshaking,VERIFY]

The input format is free format. You may exchange any two fields that will not corrupt your setup procedure. The default parameters are baud=9600, parity=N, data=8, stop=1, handshaking=disable (need not specify).

devicename

Specifies which device is control by RW device driver. To specify device name, there are two methods.

1. one bye one
   to enter each device name one by one, for example COM5, COM6, COM12.
2. group
   to enter several device name by group, for example COM5(8) means there are COM5, COM6, COM7, COM8, COM9, COM10, COM11, COM12.

baud

Baud rate. You may key in one of the follows, 38400, 19200, 9600 4800, 2400, 1200, 600, 300.

parity

parity bit.

N	none parity
E	even parity
O	odd parity

data

Data bits.

5	five data bits
6	six data bits
7	seven data bits
8	eight data bits

stop

Stop bits.

1	one stop bit
2	two stop bit

handshaking

Select software or hardware handshaking

XONXOF	enable software handshaking
XOFXON	the same as XONXOF
RTSCTS	enable RTS/CTS handshaking
CTSRTS	the same as RTSCTS
default	disable handshaking

VERIFY

List all setting parameters.

The following examples are provided to help clarify the use of the RSETUP command.

EXAMPLE E.3.1

A> RSETUP COM5

To setup I/O port COM5, the default parameters are baud rate=9600, parity=N, data bits=8, stop bits=1, disable handshaking.

EXAMPLE E.3.2

A> RSETUP 9600,COM5(4),E,8,2,XONXOF,VERIFY

To setup I/O port COM5, COM6, COM7, COM8, software handshaking protocol is enable, baud rate is 9600, parity check is

even, data bits is 8, stop bits is 2. After I/O ports are setup, system list all setting parameters.

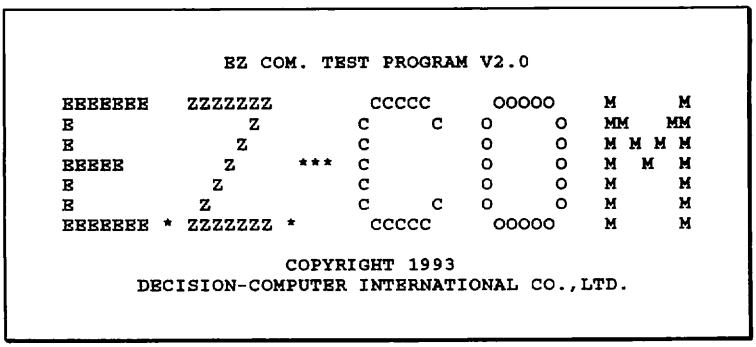
E.4 Diagnostic Under MS/DOS

The RTESTCOM program provides diagnostic routine to test your serial ports under MS/DOS operating system.

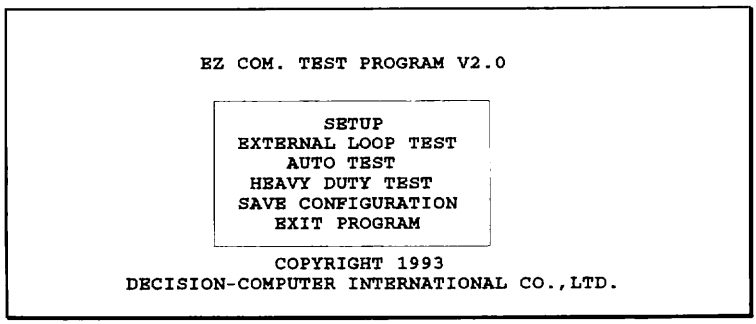
To test your serial port under MS/DOS, please type

A> RTESTCOM

then screen will display as follows



wait a moment, the screen will display as follows :



1. SETUP RS232

You may specify communication parameters such as baud rate, data bits ... etc.

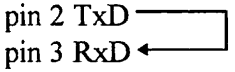
2. EXTERNAL LOOP TEST

When you select external loop test, your typing characters will be sent from one serial port to another serial port. Before you select external loop test, please connect pin 2 of send port to pin 3 of receiver port and pin 3 of send port to pin 2 of receiver port. Since the serial communication is full duplex, you can loopback send and received port in one port.

a. Connects two rs232 ports

pin 2 TxD ——————→ RxD pin 3
pin 3 RxD ←—————— TxD pin 2

b. Loopback



3. AUTO TEST

When you select auto test, the RTESTCOM will test your serial port automatically. External loop test is used to check the selected port. The sending character are read from **README.DOC** file.

4. HEAVY DUTY TEST

When you select heavy duty test, the diagnostic program will test your serial ports in a long period. The program set variable baud rate and variable block size to test your serial card. In the meanwhile, the screen will display the statistical analysis reports.

5. SAVE CONFIGURATION

Save communication parameters to disk file. When RTESTCOM is loaded, it will load the preconfiguration from disk file. The configuration file name is **RTESTCOM.CFG**.

You can use <RETURN> key to let RTESTCOM accepts your selection during you execution the RTESTCOM program, and press <ESC> key to return the last menu.

APPENDIX F PROGRAMMING TECHNIQUES UNDER MS/DOS

There are three methods to write application programs to control serial communication.

1. **Use device driver name to perform I/O.**
2. **Use EZ COM library through DOS interrupt routines.**
3. **Use low level BIOS interrupt routines.**

F.1 How to Use Device Driver

After device driver is installed, you may use device name (such as COM5, COM6, ... etc.) as a file to perform I/O operations. Since many high level languages provide several kinds of I/O operations, e.g. in C language, the write statement for low level I/O and fprintf for high level I/O. You must care which I/O routine is used by yourself to prevent the lose of I/O data, if you use high level I/O, you must use flush to dump buffer contents to serial device when necessary.

In the following, we provide BASIC, C, and PASCAL examples.

1. Loopback test (BASIC version)

```
5   REM OPEN LOGICAL DEVICE COM5
10  OPEN "O",#1,"COM5"
25  CNT=0
30  FOR I=&H31 TO &H80
40    CNT=CNT+1
50    A$=CHR$(I)
60    PRINT #1,A$;
100 NEXT I
110 PRINT CNT
180 CLOSE #1
190 FOR I = 1 TO 1000 :NEXT I
200 OPEN "I",#2,"COM5"
230 FOR I= &H31 TO &H80
250   A$=CHR$(I)
```

```
271    B$=INPUT$(1,#2)
275    PRINT "DATA ";A$, B$
280    IF A$ <> B$ THEN PRINT "ERROR"
290    IF A$ <> B$ THEN GOTO 310
300 NEXT
310 CLOSE #2
320 END
```

2. File Transfer (BASIC version) Part 1

```
20 REM SERIAL DRIVER DEMO PROGRAM FOR BASIC LANGUAGE
30 REM
40 REM THIS PROGRAM WILL TRANSMIT MESSAGES TO
50 REM REMOTE SITE THROUGH DEVICE DRIVER
60 READ A$
70 OPEN "O",#1,A$
80 FOR I = 1 TO 10
90    READ M$
100   PRINT #1,"LINE ";I,M$
110 NEXT I
120 PRINT #1,"@";
130 CLOSE #1
140 PRINT "DATA TRANSMITTED"
150 END
155 REM THE DEVICE NAME IS COM11
160 DATA "COM11"
170 DATA "SERIAL DRIVER DEMO PROGRAM FOR BASIC
LANGUAGE"
180 DATA "THIS MESSAGE IS RECEIVED FROM REMOTE SITE"
190 DATA "THIS IS LINE #3"
200 DATA "TRANSMIT OK"
210 DATA "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
220 DATA "abcdefghijklmnopqrstuvwxyz"
230 DATA "THIS DEVICE DRIVER IS VERY EASY TO USE"
240 DATA "SEE REFERENCE MANUAL FOR MORE INFORMATION"
250 DATA "COPYRIGHT BY DECISION-COMPUTER"
260 DATA "GOOD LUCK !"
```

2. File Transfer (BASIC version) Part 2

```
20 REM SERIAL DRIVER DEMO PROGRAM FOR BASIC LANGUAGE
40 REM THIS PROGRAM WILL RECEIVE MESSAGES
50 REM FROM REMOTE-SITE COMPUTER
60 READ A$
70 OPEN "I",#1,A$
80 INPUT #1,M$
90 IF M$ = "@" THEN 120
100 PRINT M$
110 GOTO 80
```

```
120 CLOSE #1
130 PRINT "DATA RECEIVED"
140 END
145 REM THE DEVICE NAME IS COM11
150 DATA "COM11"
```

3. Loopback Test (C version)

```
#include <stdio.h>
#include <stdlib.h>

void error(char *s)
{
  puts(s);
  exit(1);
}
main()
{
  int i,cnt;
  char b;
  FILE *f;

  if((f = fopen("COM5","wb")) == NULL)
  error("Device open error");
  cnt = 0;
  for(i = 0x31;i < 0x80;i++)
  {
    cnt++;
    fprintf(f,"%c",i);
  }
  printf("%d\n",cnt);
  fclose(f);
  for(i = 1;i <= 10000;i++);
  if((f = fopen("COM5","rb")) == NULL)
    error("Device open error");
  for(i = 0x31; i < 0x80;i++)
  {
    fscanf(f,"%c",&b);
    printf("DATA %c %c\n",i,b);
    if(i != b) error("ERROR");
  }
  fclose(f);
}
```

4. Clock (C version)

```
#include <stdio.h>
#include <io.h>
#include <conio.h>
```

```
#define DEVNAME "COM5"

void printdata(s)
char *s;
{
  char id[9],t1[3],t2[3],d1[3],d2[3];
  id[8] = 0;
  t1[2] = 0;
  t2[2] = 0;
  d1[2] = 0;
  d2[2] = 0;
  strncpy(id,&s[5],8);
  strncpy(t1,&s[14],2);
  strncpy(t2,&s[16],2);
  strncpy(d1,&s[18],2);
  strncpy(d2,&s[20],2);
  printf(" ID = %s TIME : %s:%s DATE : %s/%s\n",
id,t1,t2,d1,d2);
}

main()
{
  FILE *f;
  char s[128];
  int fh,count,c,z,i;

  clrscr();
  if((f = fopen(DEVNAME,"rb+")) == NULL)
  {
    puts("Open Error !");
    exit(1);
  };
  fh = fileno(f);
  do {
    c = 1;
    write(fh,&c,1);
    do {
      z = read(fh,&c,1);
    } while(z==0 && !kbhit());
    c &= 0x7f;
    if(c == 0x30)
    {
      c = 3;
      write(fh,&c,1);
      delay(100);
      count = read(fh,s,64);
      for(i=0;i<count;i++) s[i] &= 0x7f;
      printdata(s);
      c = 9;
```

```
      write(fh,&c,1);
    }
  } while(!kbhit());
}
```

5. Loopback Test (PASCAL version)

```
program DEMO;
const
  device = 'COM5';
var
  i,cnt : integer;
  a,b : char;
  f : text;
begin
  assign(f,device);
  rewrite(f);
  cnt := 0;
  for i := $31 to $80 do
  begin
    cnt := cnt + 1;
    a := chr(i);
    write(f,a);
  end;
  writeln(cnt);
  close(f);
  for i := 1 to 10000 do;
  assign(f,device);
  reset(f);
  for i := $31 to $80 do
  begin
    a := chr(i);
    read(f,b);
    writeln('DATA ',a:8,b:8);
    if a <> b then
    begin
      writeln('ERROR');
      exit;
    end;
  end;
  close(f);
end.
```

F.2 EZ COM Library

EZ COM provides C and PASCAL library routines to let user to write application programs to control serial communication via device drivers.

To write an application program, you must link RCOMLIB.C or RCOMLIB.PAS file according to your selection language. In the following, we describe the EZ COM library more details.

routine name	description
cominit	setup communication parameters
comtxch	send a character
comrxch	receive a character
comstat	check communication status
comactive	check if port is usable
comsetDTR	set DTR signal ON or OFF
compeek	check next character in queue
comsetflow	set flow control
comtxblk	send one block data
comrxblk	receive one block data
combreak	send break signal
cominstalled	check if driver installed
comsetRTS	set RTS signal ON or OFF
comsettimeout	set time out
comgettimeout	get time out
comgetbufstat	return no. of characters in queue
comgetDCD	get DCD signal
comgetDSR	get DSR signal
comgetCTS	get CTS signal
comgetflow	get flow control
comtxs	send a character string
comrxs	receive a character string
comgetinfo	get port status
comioctl	extended I/O control

1. cominit(portno, baud, db, parity, sb)

int cominit

unsigned int baud

int portno, db, parity, sb

The cominit is used to setup several communication parameters.

portno (int)

port number.

The port number are defined at MS/DOS CONFIGURATION CHAPTER.

baud (unsigned int)

baud rate. Either B38400, B19200, B9600, B4800, B2400, B1200, B600, B300.

B38400 (38400)	38400 baud
B19200 (19200)	19200 baud
B9600 (9600)	9600 baud
B4800 (4800)	4800 baud
B2400 (2400)	2400 baud
B1200 (1200)	1200 baud
B600 (600)	600 baud
B300 (300)	300 baud

db (int)

data bits. Either CS8, CS7, CS6, CS5.

CS8 (8)	8 bits
CS7 (7)	7 bits
CS6 (6)	6 bits
CS5 (5)	5 bits

parity (int)

parity bit. Either PNONE, PEVEN, PODD.

PNONE (0)	none parity check
PODD (1)	odd parity
PEVEN (2)	even parity

sb (int)

stop bits. Either ST1 or ST2.

ST1 (1)	one stop bit
ST2 (2)	two stop bits

return value (int)

0	success
1	fail

2. comtxch(portno, ch)

int comtxch
int portno
char ch

The comtxch is used to send a character to serial port.

ch (char)

transmitted character.

return value (int)

0	success
1	fail

3. comrxch(portno, ch)

int comrxch
int portno
*char *ch*

The comrxch is used to get a character.

ch (*char)

received character.

return value (int)

0	success
1	fail (receive no character)

4. comstat(portno)

int comstat
int portno

The comstat is used to get serial port status, the status indicator is shown in the following.

return value (int)

bit 15	time out error.
bit 14	transmitter shift register empty
bit 13	transmitter holding register empty

bit 8	data ready (data available in input buffer)
bit 7	received line signal detect (RLSD)
bit 5	data set ready (DSR)
bit 4	clear to send (CTS)

You may check the parameters to detect which bit is set.

S_CTS	4	0X0010
S_DSR	5	0X0020
S_DCD	7	0X0080
S_RDA	8	0X0100
S_THRE	13	0X2000
S_TSRE	14	0X4000
S_TIMEOUT	15	0X8000

5. comactive(portno)

int comactive

int portno

Check if the port is active ?

return value (int)

1	active
0	inactive

6. comsetDTR(portno, n)

void comsetDTR

int portno,n

Set DTR signal ON or OFF.

n (int)

ACTIVE (1)	set DTR ON
INACTIVE (0)	set DTR OFF

7. compeek(portno)

int compeek

int portno

The compeek is used to perform nondestructively read a character. The read character remains in the receive buffer.

return value (int)

0XFFFF	no character

?	received character

8. comsetflow(portno,n)

void comsetflow

int portno, n

To set communication protocol.

n (int)

F_XONXOF (0X09)	set XON/XOFF protocol
F_RTSCTS (0X01)	set CTS/RTS protocol

9. comtxblk(portno, pointer, n)

int comtxblk

int portno,n

*void far *pointer*

To transmit a block of data, it uses one BIOS call to transmit a block of data.

pointer (pointer)

pointer to start of block.

n (int)

block length.

return value (int)

number of characters be sent.

10. comrxblk(portno, pointer, n)

int comrxblk

int portno, n

*void far *pointer*

To receive a block of data, it uses one BIOS call to receive a block of data.

pointer (pointer)

pointer to start of block.

n (int)

number of characters should be received.

return value (int)

number of characters actually received.

11. combreak(portno, brk)

void combreak

int portno, brk

Send the break signal to driver.

brk (int)

ACTIVE (1)	break the communication
INACTIVE (0)	unbreak the communication

12. cominstalled(portno)

int cominstalled

int portno

Check if the driver is installed ?

return value (int)

1	is installed
0	not installed

13. comsetRTS(portno,n)

void comsetRTS

int portno,n

Set RTS signal ON or OFF.

n (unsigned int)

ACTIVE (1)	set RTS ON
INACTIVE (0)	set RTS OFF

14. comsettimeout(portno, timeout)

void comsettimeout

int portno,timeout

Set time out.

timeout (int)

time out number, the unit is 1/18 sec.

15. comgettimeout(portno)

int comgettimeout

int portno

Get time out.

return value (int)

The time out value, unit is 1/18 sec.

16. comgetbufstat(portno)

int comgetbufstat

int portno

The comgetbufstat is used to count the received characters in buffer.

return value (int)

count of received characters in buffer.

17. comgetDCD(portno)

int comgetDCD

int portno

Get DCD signal.

return value (int)

1	set ON
0	set OFF

18. comgetDSR(portno)

int comgetDSR

int portno

Get DSR signal.

return value (int)

1	set ON
0	set OFF

19. comgetCTS(portno)

int comgetCTS

int portno

Get CTS signal.

return value (int)

1	set ON
0	set OFF

20. comgetflow(portno, n)

int comgetflow

*int portno, *n*

To get communication protocol.

n (*int)

enable/disable hardware or software protocol.

0x09	XON/XOFF protocol
0x01	CTS/RTS protocol

return value (int)

0	success
1	fail

21. comtxs(portno, s)

int comtxs
int portno
*char *s*

The comtxs is used to send one string to serial port, it call BIOS for each character.

s (*char)

one character string to be sent.

return value (int)

0	success
1	fail

22. comrxs(portno, s)

int comtxs
int portno
*char *s*

The comrxs is used to get one string from ring buffer, it call BIOS for each character.

s (*char)

the pointer to the receive string.

return value (int)

0	success
1	fail

23. comgetinfo(portno, baud, db, parity, sb)

void comgetinfo
*unsigned int *baud*
*int portno, *db, *parity, *sb*

The comgetinfo is used to get communication parameters of each port.

baud (*unsigned int)

baud rate.

db (*int)

data bits. Either 5, 6, 7 or 8 data bits.

parity (*int)

parity bit. Either NONE, EVEN, ODD.

NONE (0)	none parity check
ODD (1)	odd parity
EVEN (2)	even parity

sb (*int)

stop bits. Either 1 or 2 stop bits.

24. comioctl(portno, ioctrl)

void comioctl

int portno

long int ioctrl

Extend I/O control. This function provides SCC protocol.

ioctrl (long int)

Four bytes command is stored in this field. For more details, please see the file "RWTECH.DOC" in the distribution diskette.

F.3 Examples

In the distribution disk, we provide file transfer and setup program, all of these programs use EZ COM library.

RSETUP.C

Setup communication parameters.

RTX.C

Transfer file.

RRX.C

Receive file.

F.4 I/O Control Via Software Interrupt

To communicate directly with device drivers, the EZ COM provides a set of subfunctions grouped under extended BIOS INT 13H. All these functions are included in the EZ COM library, however, if you want control I/O through BIOS interrupt by yourself, you need prepare CPU register contents and fork a

system call. In the following, we will describe these I/O control functions and teach users to use it.

All functions are designed as a FOSSIL driver. The specification of FOSSIL driver is shown in the "**FOSSIL.DOC**" file of distribution diskette.

Before you perform system call, you must specify subfunction number in **ah** register.

ah	description
00H	setup communication parameters
01H	send one character
02H	receive one character
03H	get serial port status
04H	activate port
06H	set DTR
0CH	nondestructive read ahead
0FH	flow control for serial I/O
18H	block read
19H	block write
1AH	break begin or end
1BH	return information about this driver
3CH	set RTS
3DH	set timeout
3EH	get status
3FH	extended I/O control

1. set up communication parameters

ah register

always set to 0h.

al register

al is used to setup communication parameters.

bit 7, 6, 5	setup baud rate
bit 4, 3	setup parity
bit 2	setup stop bits

bit 1, 0	setup data bits

bit contents

baud rate	7	6	5
9600	1	1	1
4800	1	1	0
2400	1	0	1
1200	1	0	0
600	0	1	1
300	0	1	0
38400	0	0	1
19200	0	0	0

bit contents

parity	4	3
even	1	1
odd	0	1
none	x	0

bit contents

stop bit	2
2	1
1	0

bit contents

data bit	1	0
8	1	1
7	1	0
6	0	1
5	0	0

dx register

port number.

return value (ax register)

bit 15 to bit 8 : line status

bit 7 to bit 0 : modem status

bit 15	time out error
bit 14	transmitter shift register empty

bit 13	transmitter holding register empty
bit 12	0
bit 11	0
bit 10	0
bit 9	input buffer overrun
bit 8	data available in receive buffer
bit 7	received line signal detect
bit 6	ring indicator
bit 5	data set ready
bit 4	clear to send
bit 3	always set to 1 upon return
bit 2	delta data carrier detect
bit 1	delta DSR
bit 0	delta CTS

2. send a character

ah register

always set to 1h.

al register

character.

dx register

port number.

return value (ax register)

line status and modem status.

3. receive a character

ah register

always set to 2h.

al register

received character.

dx register

port number.

return value

ah	line status
al	receive character

4. get port status

ah register

always set to 3h.

dx register

port number.

return value (ax register)

line status and modem status.

5. activate port

ah register

always set to 4h.

dx register

port number.

return value

ax	1954h if successful
bl	maximum function number supported
bh	rev of FOSSIL specification

6. set DTR

ah register

always set to 6h.

al register

0	turn off DTR
1	turn on DTR

dx register

port number.

7. nondestructive read ahead

ah register

always set to Ch.

dx register

port number.

return value

ah	0, if character is available
al	the read character
ax	FFFF, if no character is available

The character read by this operation is not removed from input queue. This means that the same character will be obtained in next read operation.

8. set flow control

ah register

always set to Fh.

al register

enable or disable flow control. When set to 1, means enable.

bit 0	enable / disable receiving XON/XOFF
bit 1	enable / disable RTS/CTS
bit 2	should be set to 1
bit 3	enable / disable transmitting XON/XOFF

dx register

port number.

9. receive block from a port

ah register

always set to 18h.

dx register

port number.

cx register

length of block.

es:di register

starting address of block.

return value (ax register)

count of received bytes.

10. send block to a port

ah register

always set to 19h.

dx register

port number.

cx register

number of bytes to be sent.

es:di register

starting address for received block.

return value (cx register)

count of sent bytes.

11. send break signal

ah register

always set to 1Ah.

dx register

port number.

al register

1	send break signal
0	stop to send break signal

12. return information about driver

ah register

always set to 1Bh.

dx register

port number.

cx register

length of block.

es:di register

starting address of block.

return value (ax register)

count of received bytes.

13. turn on RTS

ah register

always set to 3Ch.

dx register

port number.

al register

0	turn off RTS
1	turn on RTS

14. set timeout

ah register

always set to 3Dh.

dx register

port number.

al register

timeout no. each unit is 1/18 sec.

15. return status

ah register

always set to 3Eh.

dx register

port number.

al register

0	get timeout (1/18 sec)
1	get Rx characters in queue
2	get DCD
3	get DSR
4	get CTS
5	get DTR
6	get RTS
7	get line status (box)
8	box version
9	buffer full on remote site
10	line spec or baud control
11	flow control

return (ax register)

0 : inactive, otherwise active.

number of characters in the queue.

the functions with al = 10 and al = 11 are the same as subfunctions with ah = 0h and ah = 0fh. (please see item 1 and 8)

16. extended I/O control

ah register

always set to 3Fh.

dx register

port number.

al register

function number.

cx register

Specifies interrupt level of serial port.

The syntactic specification of SETUP is

SETUP portaddr,[BAUD=baud,parity,data,stop, IRQ=no,MCR=d2,handshaking,VERIFY]

The input format is free format. You may exchange any two fields that will not corrupt your setup procedure. The default parameters are baud=9600, parity=N, data=8, stop=1, no=4, d2=3, handshaking=XOFF.

portaddr

Specifies which port address is control by PCCOM device driver. You may input 3F8 or 2F8.

BAUD=baud

Baud rate. You may key in any baud rate, the device driver will adjust your baud rate to desire baud rate that the serial port can accept. If you key in 65535 baud, this means 115200. According to our test, The PCCOM device driver can accept up to 115200 baud under 8m cable. You need not key in BAUD= when baud rate matches the follows, 9600, 4800, 2400, 1200, 600, 300, 150.

parity

parity bit.

N	none parity
E	even parity
O	odd parity

data

Data bits.

7	seven data bits
8	eight data bits

stop

Stop bits.

1	one stop bit
2	two stop bits

IRQ=no

Select which interrupt is used. The interrupt number must matches the interrupt number of PCCOM.COM installation. When a character is received from anyone of portadd, then the IRQ is emerged, which active the corresponding device driver that has the same IRQ specification.

MCR=d2

Select OUT1 and OUT2 signal.
d2=3 means OUT1=0 and OUT2=0,
d2=7 means OUT1=1 and OUT2=0,
d2=B means OUT1=0 and OUT2=1,
d2=F means OUT1=1 and OUT2=1.

handshaking

Select software handshaking.

XON	enable software handshaking
XOFF	disable software handshaking

VERIFY

List all setting parameters. After serial ports are setup, system will check whether the specified port is good, if the corresponding port is bad, system will prompt ???? to notice user.

The interrupt setting and port address of standard COM1 and COM2 are shown in the following.

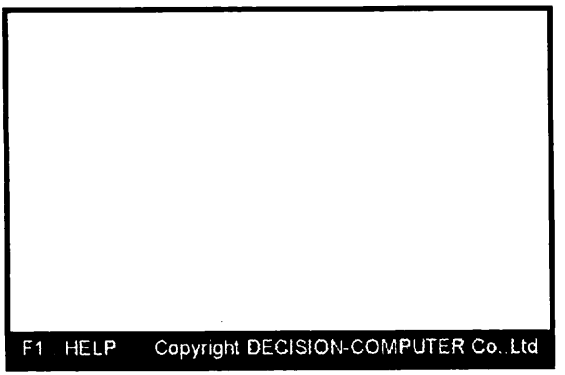
COM1 address 3F8
COM2 address 2F8
interrupt for COM1 IRQ4
interrupt for COM2 IRQ3
modem control register B

To setup COM1, the commands are shown in the following.

PCCOM /4
SETUP 3F8,MCR=B,XON,VERIFY

G.4 How to Execute VT.EXE

After PCCOM and SETUP are executed, just run VT to enter virtual terminal evaluation mode. The screen you will see is shown below.



You may key in character from keyboard to communicate with host computer or select function to setup communication parameters. All available function keys are shown in the following.

(F1) *HELP*

When press F1 key, system will display functions that you can select.

(F2) *S/CTRL*

Press F2 key will send XOFF code to host computer to notice that virtual terminal is too busy. When you want set XON to host computer again, please press Q/CTRL.

(F3) *SETUP*

Setup the communication protocol such as baud rate, data bits, stop bits ... etc.

(F4) *C/CTRL*

Press F4 key will send C/CTRL code to remote computer to kill the executing process.

(F5) *LISTCFG*

List the communication parameters on the screen.

(F6) *SAVECFG*

Save the setup communication parameters to disk file named VT.CFG.

(F7) *Download (receive data from host)*

Receive data from the remote site then save these data to a user specified file.

(F8) *Upload (transmit data to remote site)*

Transfer a user specified file that located at PC (which is now run VT) to remote site.

(F9) *EXIT*

Exit virtual terminal evaluator.

(F10) *SHELL*

Fork MS/DOS shell command interpreter, the terminal evaluator is still resident on the main memory. When you enter shell command, you may run any MS/DOS commands and programs, because you stay at MS/DOS mode. To enter VT again, please type EXIT.

G.4.1 Download Data from Host

When you select F7, system will ask a file name which is used to save data that sent from remote computer.

Suppose your virtual terminal is connected to XENIX operating system and you want download a file aa from host to PC, and save it in PC.DOC file, the procedure is

cat aa	*(don't enter <CR>)*
(F7)	*(select F7)*
PC.DOC	*(enter file name)*
(<CR>)	

G.4.2 Upload Data to Host

When you select F8, system will ask a file name and send the file contents to host computer. Under XENIX operating system, suppose you want send a file PC.DOC from PC to host computer and save it in aa file, the upload procedure is

cat > aa
(F8)
PC.DOC
(<CR>)

G.4.3 Setup

When you select F3, you may modify the communication parameters. After communication parameters are setup, you may use F6 to save the configuration and F5 to list the configuration. When VT is loaded, it will check VT.CFG to load the last save configuration automatically. The communication parameters are list in the following.

1. IRQ number
Select which interrupt.

2. port number
Select which I/O port.

3. baud rate
Select communication speed.

4. data bits
Select data bits

5. stop bits
Select stop bits.

6. parity
Press SPACE bar to select parity.

7. ECHO
Press SPACE bar to select local ECHO ON/OFF. When your host computer has ability to send back the received

character, you must select local ECHO OFF to let host computer prompt your typing character.

8. new line

Press SPACE bar to select new line ON/OFF. VT will add <CR> code when it receive <LF> at new line ON mode.

9. delay time

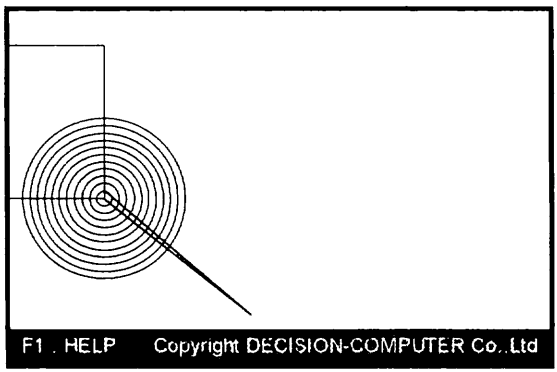
Select delay time interval between two transmitting characters when upload is processing.

G.5 Examples

The following control string will draw a figure which is shown in the following. Don't forget to press any key to enter text mode.

```
<ESC>P
L00000000000010000000
L00100000000010000100
L00100001000000000100
L00000001000000000000
P0005000030
P0005200032
P0005400034
P0005600036
P0005800038
P0006000040
P0006200042
P0006400044
P0006600046
P0006800048
C001000010000050
C001000010000040
C001000010000030
C001000010000020
C001000010000010
C001000010000005
C001000010000015
C001000010000025
C001000010000035
C001000010000045
C001000010000055
M0010000100
D0030000200
D0009000090
Q
```

The graphics control string is stored in the **VT.DOC** file.



APPENDIX H TRANSPARENT PRINTING UNDER UNIX

Under UNIX operating system, the lpcontrol command can be used to provide transparent printing function on terminal. In the following, we will introduce how to use the transparent printing utilities.

H.1 Device Name

The device name of transparent printing on terminal are /dev/rwlpCBP, where C is a letter from "a" to "d" corresponds to four adapters, B is a number from 1 to 4 corresponds to four terminal concentrator, and P is a number from 1 to 8 corresponds to eight ports. For example, the /dev/rwlpa11 is the device name for first card + first box + first port.

C	card number
B	terminal concentrator number
P	port number

The printing entry is associated with the terminal entry (tty), so that you can not use stty command to change communication parameter (such as baud rate, flow control) from /dev/rwlpCBP. If you need change communication parameter, please use stty to change /dev/ttyXYZ (see appendix D). For example, you can not use

stty 9600 </dev/rwlpa11
you must use stty 9600 </dev/ttya11.

H.2 Transparent printing control

The lpcontrol command is used to control transparent printing, its command syntax is defined as

lpcontrol COMMAND PORT [parameter]

1. PORT

PORT value can be 1 to 128. For example /dev/rwlpa11 is 1, /dev/rwlpa12 is 2, /dev/rwlpa21 is 9.

2. COMMAND

a. GETSEQ PORT

Read the PRINT ON/OFF control sequence defined on the port. For example, read the information from port 2.

lpcontrol GETSEQ 2

b. SETSEQ PORT PRINTON PRINTOFF

Set the PRINT ON/OFF control sequence. For example, if you want set PRINT ON by ESC[5i and PRINT OFF by ESC[4i (where ESC is escape code) to port 1, you can use

lpcontrol SETSEQ 1 "\033[5i" "\033[4i"

The default ANSI sequence is set by ESC[5i for PRINT ON and ESC[4i for PRINT OFF.

3. SETRATIO PORT RATIO

To set the printer speed. It defines the ratio time used by the printer on the port. It depends on the buffer size and the speed of the printer. Default RATIO is 40, maximum value is 256. If you increase the value, you will allocate more time for a slowly printer. For example, to set port 1 with a ratio of 55,

lpcontrol SETRATIO 1 55

4. GETRATIO PORT

To get the current ratio from the PORT.

H.3 How to Use Transparent Printer

If you want to start a printing on the first card, first box and the first port, you can use the cat command.

cat filename >/dev/rwlpa11

You can include the printer inside the printing spooler. If you use the name "YP" to start a printing, you can use

lp filename -d YP

or

LPDEST=YP

export YP

lp filename

APPENDIX I TECHNICAL REFERENCE INFORMATION

The technical reference of EZ COM intelligent 8 port card is shown in the distribution diskette, you can type the file named "**RWTECH.DOC**" for more details. The specification of FOSSIL driver is also in the "**FOSSIL.DOC**" file.

APPENDIX J IMPORTANT TIPS

For ESDI, SCSI users in xenix, UNIX system. If you use ESDI or SCSI hard disk controller for xenix or UNIX system, probably you have the hard disk drivers supplied by manufacturer. Please make sure that the ESDI or SCSI device driver files are resident in your system before installing EZ COM intelligent card device driver. In xenix system, you can check /usr/sys/conf or /usr/sys/io directories for ESDI or SCSI device driver files. In UNIX system, please check /etc/conf/pack.d directory for these files. If you can not find these files in your system, please contact the ESDI or SCSI controller supplier to get these files.

APPENDIX K WARRANTY INFORMATION

K.1 Copyright

Copyright 1993 DECISION COMPUTER INTERNATIONAL CO., LTD. all rights reserved. No part of EZ COM software and manual may be reproduced, transmitted, transcribed, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written permission of DECISION COMPUTER INTERNATIONAL CO. LTD.

Each piece of EZ COM package permits user to use EZ COM only on a single computer, a registered user may use the program on a different computer, but may not use the program on more than one computer at the same time.

Corporate licensing agreements allow duplication and distribution of specific number of copies within the licensed institution. Duplication of multiple copies is not allowed except through execution of a licensing agreement. Welcome call for details.

K.2 Warranty Information

EZ COM warrants that for a period of one year from the date of purchase (unless otherwise specified in the warranty card) that the goods supplied will perform according to the specifications defined in the user manual. Furthermore that the EZ COM product will be supplied free from defects in materials and workmanship and be fully functional under normal usage.

In the event of the failure of a EZ COM product within the specified warranty period, EZ COM will, at its option, replace or repair the item at no additional charge. This limited warranty does not cover damage resulting from incorrect use, electrical interference, accident, or modification of the product.

All goods returned for warranty repair must have the serial number intact. Goods without serial numbers attached will not be covered by the warranty.

Transportation costs for goods returned must be paid by the purchaser. Repaired goods will be dispatched at the expense of EZ COM.

To ensure that your EZ COM product is covered by the warranty provisions, it is necessary that you return the Warranty card.

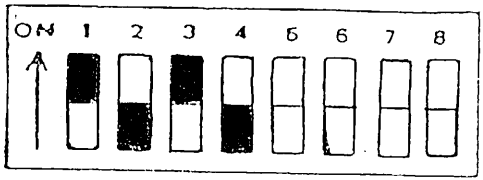
Under this Limited Warranty, EZ COM's obligations will be limited to repair or replacement only, of goods found to be defective as specified above during the warranty period. EZ COM is not liable to the purchaser for any damages or losses of any kind, through the use of, or inability to use, the EZ COM product.

EZ COM reserves the right to determine what constitutes warranty repair or replacement.

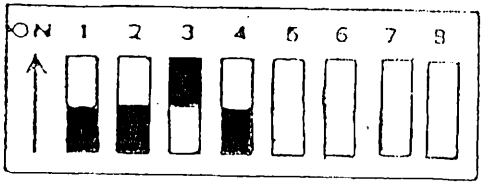
Return Authorization: It is necessary that any returned goods are clearly marked with an RA number that has been issued by EZ COM. Goods returned without this authorization will not be attended to.

EASY COM SWITCH SETTING

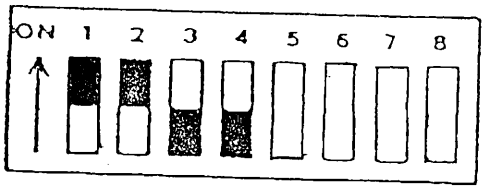
0A000:0000 (SW. 1,3 ON)



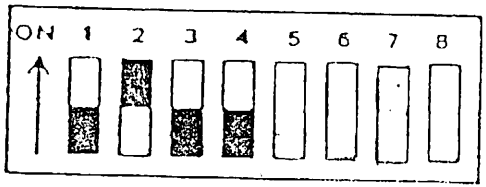
0B000:0000 (SW. 3 ON)



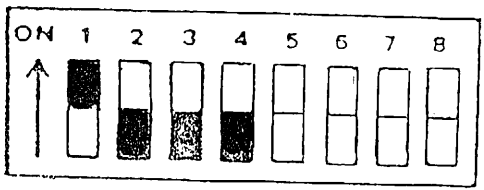
0C000:0000 (SW. 1, 2 ON)



0D000:0000 (SW. 2 ON)



0E000:0000 (SW. 1 ON)



0E000:0000 (SW. ALL OFF)